The Grid File I/O (Gfio) Library Version 1.0.8

ROB LUCCHESI AND ARLINDO DA SILVA Data Assimilation Office, NASA/GSFC, Greenbelt, MD 20771

August 23, 1999 (Original design October 1997)

Contents

1	System Overview		
2	Routine/Function Prologues		
	2.1	GFIO_Create - Creates a DAO gridded file for writing (Source File: gfio.F)	4
	2.2	GFIO_Open - Opens an existing DAO gridded file (Source File: gfio.F)	6
	2.3	GFIO_PutVar - Write a variable to the file (Source File: gfio.F)	7
	2.4	GFIO_GetVar - Read a variable from the file (Source File: gfio.F)	8
	2.5	GFIO_DimInquire – Gets dimension information from a GFIO file. (Source	
		File: gfio.F)	10
	2.6	GFIO_Inquire - Get information about a GFIO file. (Source File: gfio.F) .	11
	2.7	GFIO_Close - Closes file (Source File: gfio.F)	13
	2.8	GFIO_PutIntAtt – Write a user-defined integer attribute (Source File: gfio.F)	14
	2.9	GFIO_PutRealAtt - Write a user-defined real attribute (Source File: gfio.F)	15
	2.10	GFIO_PutCharAtt – Write a user-defined character attribute (Source File:	
		$\operatorname{gfio.F})$	16
	2.11	GFIO_GetAttNames – Get global attribute names (Source File: gfio.F)	16
	2.12	GFIO_AttInquire – Get information about an attribute (Source File: gfio.F)	17
	2.13	GFIO_GetIntAtt – Read a user-defined integer attribute (Source File: gfio.F)	18
	2.14	GFIO_GetRealAtt – Read a user-defined real attribute (Source File: gfio.F)	19
	2.15	GFIO_GetCharAtt - Read a user-defined character attribute (Source File:	
		gfio.F)	20

1 System Overview

Basic Requirements:

- (1) Design an interface for writing HDF format data files from the June 1998 GEOS-3 production system without requiring the direct insertion of HDF Toolkit calls in the code.
- (2) Design this interface to be flexible enough to support usage in other DAO applications that read or write HDF data.
- (3) Output files should conform the COARDS conventions. This allows the data to be immediately usable by GrADS, other visualization packages and utilites such as ncdump.
- (4) Provide a library that is callable from a Fortran 77 application with a portable interface.
- (5) The library must also be callable by C, perhaps with the use of a tool like Cfortran.h.

The primary motivation behind GFIO is to provide an easy way for the GEOS-DAS to write HDF format data while hiding calls to the HDF libraries. Additionally, it is hoped that this library will be of general use for reading or writing HDF files in applications other than the GEOS-DAS.

The typical calling sequence for creating a file would be:

```
GFIO_Create(...)

GFIO_PutVar(...)

GFIO_PutVar(...)

...

GFIO_Close(...)

One could subsequently open the file for more writing with:

GFIO_Open(...)
NOTES:
```

- * Surface data is permitted in the same file as upper air data, however all upper air data must be defined with the same number of levels but it is not necessary to write data for each defined level. In the case that data is not written for a given level, HDF will put fills.
- * Packing is not yet implemented.
- * The time increment cannot be defined as 0, even if only writing one time.
- * Files are written using the NetCDF interface provided in the HDF library. The files conform to the COARDS conventions, meaning they have specific metadata defined by the convention.

2 Routine/Function Prologues

2.1 GFIO_Create - Creates a DAO gridded file for writing (Source File: gfio.F)

This routine is used to open a new file for a GFIO stream. Packing is not yet supported. Information about each opened stream is stored in a COMMON block contained in gfio.h. This information is later used by GFIO_PutVar. GFIO_Open should be used to open an existing file for reading or writing.

INTERFACE:

```
subroutine GFIO_Create ( fname, title, source, contact, amiss,

& im, jm, km, lon, lat, levs, levunits,

& yyyymmdd_beg, hhmmss_beg, timinc,

nvars, vname, vtitle, vunits, kmvar,

valid_range, packing_range, prec,

fid, rc )
```

USES:

Implicit NONE
include "netcdf.inc"
include "gfio.h"

INPUT PARAMETERS:

		! Global Metadata	
character*(*)	fname	! File name	
character*(*)	title	! A title for the data set	
character*(*)	source	! Source of data, e.g. NASA/DAO	
character*(*)	contact	! Who to contact about the data set, e.g.,	
		! 'Contact data@dao.gsfc.nasa.gov'	
real	amiss	! Missing value such as 1.0E15	
		! Dimension Metadata	
integer	im	! size of longitudinal dimension	
integer	jm	! size of latitudinal dimension	
integer	km	! size of vertical dimension	
		! (surface only=1)	
real	lon(im)	! longitude of center of gridbox in	
		! degrees east of Greenwich (can be	
		! -180 -> 180 or 0 -> 360)	
real	lat(jm)	! latitude of center of gridbox in	
		! degrees north of equator	
real	levs(km)	! Level (units given by levunits) of	
		! center of gridbox	
character*(*)	levunits	! units of level dimension, e.g.,	
		! "millibar", "hPa", or "sigma_level"	
integer	yyyymmdd_beg		
integer	hhmmss_beg	! First hour-minute-second to be written	

```
integer
                timinc
                              ! Increment between output times (HHMMSS)
                              ! ----- Variable Metadata -----
                              ! number of variables in file
integer
                nvars
character*(*)
                vname(nvars) ! variable short name, e.g., "hght"
character*(*)
                vtitle(nvars) ! variable long name, e.g.,
                                  "Geopotential Height"
character*(*)
                vunits(nvars) ! variable units, e.g., "meter/second"
                kmvar(nvars) ! number of levels for variable; it can
integer
                              ! either be 0 (2-D fields) or equal to km
real
        valid_range(2,nvars) ! Variable valid range; GFIO_PutVar
                              ! will return a non-fatal error if a value is
                              ! outside of this range. IMPORTANT: If packing
                              ! is not desired for a given variable, YOU MUST
                              ! set both components of valid_range to amiss.
                              ! ----- Packing Metadata ----
      packing_range(2,nvars) ! Packing range to be used for 16-bit packing
real
                              ! of each variable. IMPORTANT: If packing is not
                              ! desired for a given variable, YOU MUST set both
                              ! components of packing_range to amiss.
                              ! NOTE:
                              ! * The packing algorithm sets all values
                                   outside the packing range to missing.
                              ! * The larger the packing range, the greater
                                   the loss of precision.
integer
               prec
                              ! Desired precision of data:
                                  0 = 32 \text{ bit}
                                  1 = 64 \text{ bit}
                                  NOTE: mixing precision in the same
                                  * Mixing 32 and 64 bit precision in the
                              ļ.
                                     same file is not supported.
                                  * If packing is turned on for a variable,
                                     the prec flag is ignored.
```

```
! File handle
integer
              fid
integer
                      ! Error return code:
              rc
                      ! rc = 0
                               all is well
                        rc = -1 time increment is 0
                      ! rc = -18 incorrect time increment
                        NetCDF Errors
                        -----
                      ! rc = -30 error from nccre (file create)
                      ! rc = -31 error from ncddef
                      ! rc = -32 error from ncvdef (dimension variable)
                      ! rc = -33 error from ncaptc (dimension attribute)
```

```
! rc = -34 error from ncvdef (variable)
! rc = -35 error from ncaptc (variable attribute)
! rc = -36 error from ncaptc/ncapt (global attribute)
! rc = -37 error from ncendf
! rc = -38 error from ncvpt (dimension variable)
```

1997.09.13	da Silva/Lucchesi	Initial interface design.
1997.09.22	Lucchesi	Added timinc to interface.
1998.02.10	Lucchesi	Added support for applications running with
		64-bit reals.
1998.02.17	Lucchesi	Added time_inc, begin_time, and begin_date
		attributes to the time dimension.
1998.03.30	Lucchesi	Documentation expanded. Clean-up of code.
1998.07.07	Lucchesi	Removed vids from argument list
1998.07.09	Lucchesi	Converted timinc to seconds before saving
1998.10.09	Lucchesi	Precision flag, documentation changes.
1998.10.27	Lucchesi	Added support for packing and range checks
1998.11.18	Lucchesi	Modified timinc to be HHMMSS as given by user
1999.01.04	Lucchesi	Changed variable initialization
1999.03.30	Lucchesi	Added 'positive=down' attribute to lev.

2.2 GFIO_Open - Opens an existing DAO gridded file (Source File: gfio.F)

This routine opens an existing DAO gridded file. The file mode will be read/write. If the application already knows the contents of the file, it may begin interaction with the file using the returned file handle. Otherwise, the file handle can be used with the "inquire" routines to gather information about the contents. A negative return code indicates there were problems opening the file.

INTERFACE:

USES:

```
subroutine GFIO_Open ( fname, fmode, fid, rc )
```

Implicit NONE
include "netcdf.inc"
include "gfio.h"

INPUT PARAMETERS:

```
character*(*) fname ! File name
integer fmode ! File mode:
```

```
! O for READ-WRITE
! non-zero for READ-ONLY
```

```
integer fid ! File handle
integer rc ! Error return code:
! rc = 0 All is well
! rc = -39 error from ncopn (file open)
```

REVISION HISTORY:

1998.07.02	Lucchesi	Initial interface design.
1998.07.07	Lucchesi	Initial coding.
1998.12.09	Lucchesi	Corrected for ncopn bug.

2.3 GFIO_PutVar - Write a variable to the file (Source File: gfio.F)

This routine is used to write a variable to an open GFIO stream. Multiple vertical levels can be written at one time provided they are contiguous in memory. Date and time must be consistent with the time increment and the starting date/time as defined in GFIO_Create. Times must fall on minute boundaries to allow GrADS to work. Error checking is done for dimensions that are out of bounds.

INTERFACE:

```
subroutine GFIO_PutVar ( fid, vname, yyyymmdd, hhmmss,
& im, jm, kbeg, kount, grid,
& rc )
```

USES:

```
Implicit NONE
include "netcdf.inc"
include "gfio.h"
```

INPUT PARAMETERS:

```
fid
                                    ! File handle
integer
                                    ! Variable name
character*(*)
              vname
                                    ! Year-month-day, e.g., 19971003
integer
               yyyymmdd
                                    ! Hour-minute-second, e.g., 120000
integer
               hhmmss
                                    ! size of longitudinal dimension
integer
                im
                                    ! size of latitudinal dimension
integer
                jm
                                    ! first level to write; if 2-D grid
integer
                kbeg
                                        use kbeg = 0.
                                    ! number of levels to write
integer
                kount
                grid(im,jm,kount)
                                   ! Gridded data to write at this time
real
```

```
integer
              rc ! Error return code:
                  ! rc = 0 all is well
                  ! rc = -2 time is inconsistent with increment
                  ! rc = -3 number of levels is incompatible with file
                  ! rc = -4 im is incompatible with file
                  ! rc = -5 jm is incompatible with file
                  ! rc = -6 time must fall on a minute boundary
                  ! rc = -7 error in diffdate
                  ! rc = -12 error determining default precision
                  ! rc = -13 error determining variable type
                  ! rc = -15 data outside of valid range
                  ! rc = -16 data outside of packing range
                  ! rc = -17 data outside of pack and valid range
                  ! NetCDF Errors
                    -----
                  ! rc = -38 error from ncvpt (dimension variable)
                  ! rc = -40 error from ncvid
                  ! rc = -41 error from ncdid or ncding (lat or lon)
                  ! rc = -42 error from ncdid or ncding (lev)
                  ! rc = -43 error from ncvid (time variable)
                  ! rc = -44 error from ncagt (time attribute)
                  ! rc = -45 error from ncvpt
                  ! rc = -46 error from ncvgt
                  ! rc = -52 error from neving
                  ! rc = -53 error from ncagtc/ncagt
```

REVISION HISTORY:

1997.10.13 da Silva/Lucchesi	Initial interface design.
1998.02.10 Lucchesi	Added support for applications running with
	64-bit reals.
1998.03.30 Lucchesi	Documentation expanded. Clean-up of code.
1998.07.02 Lucchesi	Replaced vid with vname in argument list $\&$
	made related mods to code.
1998.09.24 Lucchesi	Changed error codes, removed DIM_CHECK if-def
1998.10.27 Lucchesi	Added support for packing and range checks
1998.12.15 Lucchesi	Added support for skipping times (allTimes)
1999.01.04 Lucchesi	Fixed bug in skipping times (allTimes)/also
	changed variable initialization.
1999.07.13 Lucchesi	Changes for REAL or INT time dimension

2.4 GFIO_GetVar - Read a variable from the file (Source File: gfio.F)

This routine will read one or more levels of "vname" into the buffer passed in as "grid." "fid" is the file handle returned by Gfio_open.

INTERFACE:

USES:

Implicit NONE
include "netcdf.inc"
include "gfio.h"

INPUT PARAMETERS:

```
integer
             fid
                             ! File handle
character*(*) vname
                             ! Variable name
                            ! Year-month-day, e.g., 19971003
integer
             yyyymmdd
                           ! Hour-minute-second, e.g., 120000
integer
              hhmmss
                             ! size of longitudinal dimension
integer
             im
integer
                             ! size of latitudinal dimension
             jm
             kbeg
                            ! first level to read; if 2-D grid
integer
                             ! set kbeg = 0.
                             ! number of levels to read
integer
              kount
```

```
grid(im,jm,kount) ! Gridded data read for this time
real
                  ! Error return code:
integer rc
                  ! rc = 0 all is well
                  ! rc = -2 time is inconsistent with increment
                  ! rc = -3 number of levels is incompatible with file
                  ! rc = -4 im is incompatible with file
                  ! rc = -5 jm is incompatible with file
                  ! rc = -6 time must fall on a minute boundary
                  ! rc = -7 error in diffdate
                  ! rc = -12 error determining default precision
                  ! rc = -13 error determining variable type
                  ! rc = -19 unable to identify coordinate variable
                  ! NetCDF Errors
                  ! -----
                  ! rc = -38 error from ncvpt (dimension variable)
                  ! rc = -40 error from ncvid
                  ! rc = -41 error from ncdid or ncding (lat or lon)
                  ! rc = -42 error from ncdid or ncding (lev)
                  ! rc = -43 error from ncvid (time variable)
                  ! rc = -44 error from ncagt (time attribute)
                  ! rc = -46 error from ncvgt
                  ! rc = -48 error from ncing
                  ! rc = -52 error from ncvinq
```

```
1997.10.13 da Silva/Lucchesi Initial interface design.
1998.07.07 Lucchesi Combined two GetVar routines into this one.
1998.09.24 Lucchesi Updated error codes.
1999.06.21 Lucchesi Bug fixed. Unable to read HDF-EOS files
because was still looking for "lon" and "lat"
1999.06.21 Lucchesi Added a check for time too early.
```

2.5 GFIO_DimInquire – Gets dimension information from a GFIO file. (Source File: gfio.F)

This routine is used to get dimension information from an existing GFIO file. This dimension information can subsequently be used to allocate arrays for reading data from the file. For more complete information about the contents of a file, Gfio-Inquire should be used.

INTERFACE:

```
subroutine GFIO_DimInquire (fid,im,jm,km,lm,nvars,ngatts,rc)
```

USES:

```
Implicit NONE
include "netcdf.inc"
include "gfio.h"
```

INPUT PARAMETERS:

```
integer fid ! File handle
```

OUTPUT PARAMETERS:

```
! Size of longitudinal dimension
integer
                  ! Size of latitudinal dimension
integer
           jm
integer
                  ! Size of vertical dimension
           km
                      km=0 if surface-only file
integer
                  ! Number of times
           l m
integer
           nvars ! Number of variables
integer
           ngatts! Number of global attributes
                  ! Error return code:
integer
                               all is well
                  ! rc = -19 unable to identify coordinate variable
                  ! NetCDF Errors
                     -----
                  ! rc = -40 error from ncvid
                  ! rc = -41 error from ncdid or ncding (lat or lon)
                  ! rc = -42 error from ncdid or ncding (lev)
                  ! rc = -43 error from ncvid (time variable)
```

! rc = -47 error from ncdid or ncding (time)

```
! rc = -48 error from ncinq
! rc = -53 error from ncagtc/ncagt
```

1998.07.02	Lucchesi	Initial interface design.
1998.08.05	Lucchesi	Added "ngatts"
1998.09.24	Lucchesi	Revamped error codes
1998.12.22	Lucchesi	Added IdentifyDim and associated code
1999.01.04	Lucchesi	Changed variable initialization

2.6GFIO_Inquire - Get information about a GFIO file. (Source File: gfio.F)

This routine is used to get as much information as possible about the contents of a GFIO file. The file handle (fid) is passed in and detailed information about dimensions and variables are returned to the application. A simpler inquire routine for dimension information is Gfio_DimInquire.

INTERFACE:

```
subroutine GFIO_Inquire (fid, im, jm, km, lm, nvars,
                         title, source, contact, amiss,
&
                         lon, lat, levs, levunits,
&
&
                         yyyymmdd, hhmmss, timinc,
                         vname, vtitle, vunits, kmvar,
&
&
                         valid_range , packing_range, rc)
```

USES:

```
Implicit NONE
include "netcdf.inc"
include "gfio.h"
```

INPUT PARAMETERS:

```
! ----- Global Metadata -----
      integer
                                    ! File handle from GFIO_open
INPUT/OUTPUT PARAMETERS:
      integer
                                    ! size of longitudinal dimension
                      im
      integer
                                    ! size of latitudinal dimension
                      jm
                                    ! size of vertical
                                                           dimension
      integer
                      km
      integer
                      1 m
                                    ! size of time
                                                           dimension
```

! On input, (im, jm, km, lm) contains the

! size of arrays (lon, lat, lev, yyyymmdd)

```
! as declared in the calling program.
                                     ! On output, (im,jm,km,lm) contains the
                                     ! size of the coordinate variables
                                     ! (lon, lat, lev, yyyymmdd) on file.
                                     ! number of variables on file
      integer
                      nvars
OUTPUT PARAMETERS:
      character*(*)
                                     ! Title of the data set
                      title
      character*(*)
                      source
                                     ! Where it came from
      character*(*)
                      contact
                                     ! Who to contact about the data set
      real
                      amiss
                                     ! Missing value
                                     ! ----- Dimension Metadata -----
                                     ! longitude of center of gridbox in
      real
                      lon(im)
                                     ! degrees east of Greenwich (can be
                                     ! -180 -> 180 or 0 -> 360)
                                     ! latitude of center of gridbox in
      real
                      lat(jm)
                                     ! degrees north of equator
                                     ! Level (units given by levunits) of
      real
                      levs(km)
                                         center of gridbox
      character*(*)
                                     ! units of level dimension, e.g.,
                      levunits
                                         "hPa", "sigma_level"
                                     ! Year-month-day on file
      integer
                      yyyymmdd(lm)
                       hhmmss(lm)
                                     ! Hour-minute-second on file
      integer
      integer
                         timinc
                                     ! Time increment.
                                     ! ----- Variable Metadata -----
                      vname(nvars) ! variable short name, e.g., "hght"
      character*(*)
      character*(*)
                     vtitle(nvars) ! variable long name, e.g.,
                                         "Geopotential Height"
      character*(*)
                      vunits(nvars) ! variable units, e.g., "meter/second"
      integer
                      kmvar(nvars) ! number of levels for variable; it can
                                     ! either be 0 (2-D fields) or equal to km
              valid_range(2,nvars)
                                    ! Variable valid range; set to
      real
                                        "amiss" if not known.
                                     ! ----- Packing Metadata ----
      real packing_range(2,nvars)
                                     ! Variable packing range used
                                     ! for 16-bit packing. If packing was not
                                     ! used then returned values will be amiss.
                                     ! NOTE: all unpacking is done transparently
                                             by GFIO_GetVar().
      integer
                          ! Error return code:
                 rc
                                       all is well
                              rc = -3 number of levels is incompatible with file
```

```
rc = -4 im is incompatible with file
   rc = -5 jm is incompatible with file
   rc = -8 lm is incompatible with file
   rc = -9 nvars is incompatible with file
! rc = -14 error in getdate
  rc = -20 vname strings too short
! NetCDF Errors
! -----
! rc = -41 error from ncdid or ncding (lat or lon)
! rc = -42 error from ncdid or ncding (lev)
! rc = -43 error from ncvid (time variable)
! rc = -47 error from ncdid or ncding (time)
! rc = -48 error from ncinq
! rc = -50 error from ncagtc (level attribute)
! rc = -51 error from ncagtc/ncagt (global attribute)
! rc = -52 error from ncvinq
! rc = -53 error from ncagtc/ncagt
```

!FUTURE ENHANCEMENT:

Next release should include a flag for precision.

REVISION HISTORY:

```
1998.07.02 Lucchesi Initial interface design.
1998.07.17 Lucchesi Initial coding.
1998.10.09 Lucchesi Restructured return codes.
1998.12.24 Lucchesi Modified to read non-GFIO files.
1999.01.04 Lucchesi Changed variable initialization
1999.07.13 Lucchesi Changes for REAL or INT time dimension
```

2.7 GFIO_Close - Closes file (Source File: gfio.F)

This routine is used to close an open GFIO stream.

INTERFACE:

```
subroutine GFIO_Close ( fid, rc )
```

USES:

```
Implicit NONE
include "netcdf.inc"
include "gfio.h"
```

INPUT PARAMETERS:

integer fid ! File handle

```
integer rc ! Error return code:

! rc = 0 all is well
! NetCDF Errors
! ------
! rc = -54 error from ncclos (file close)

REVISION HISTORY:

1997.10.13 da Silva/Lucchesi Initial interface design.
1998.03.30 Lucchesi Documentation expanded. Clean-up of code.
Added rc.
```

2.8 GFIO_PutIntAtt - Write a user-defined integer attribute (Source File: gfio.F)

This routine allows the user to define an integer attribute in an open GFIO file.

INTERFACE:

```
subroutine GFIO_PutIntAtt ( fid, name, count, buf, prec, rc )

USES:

Implicit NONE
  include "netcdf.inc"
  include "gfio.h"
```

INPUT PARAMETERS:

```
integer rc ! Error return code:
    ! rc = 0 all is well
! rc = -12 error determining default precision
! NetCDF Errors
! -------
! rc = -36 error from ncaptc/ncapt (global attribute)
! rc = -55 error from ncredf (enter define mode)
! rc = -56 error from ncedf (exit define mode)
```

```
1998.07.30LucchesiInitial interface design.1998.07.30LucchesiInitial coding.1998.09.24LucchesiChanged error handling.1998.09.28LucchesiAdded support for multiple precisions
```

2.9 GFIO_PutRealAtt – Write a user-defined real attribute (Source File: gfio.F)

This routine allows the user to define a real attribute in an open GFIO file.

INTERFACE:

```
subroutine GFIO_PutRealAtt ( fid, name, count, buf, prec, rc ) \label{eq:USES:} USES:
```

Implicit NONE
include "netcdf.inc"
include "gfio.h"

INPUT PARAMETERS:

OUTPUT PARAMETERS:

```
integer rc ! Error return code:
! rc = 0 all is well
! rc = -12 error determining default precision
! NetCDF Errors
! -------
! rc = -36 error from ncaptc/ncapt (global attribute)
! rc = -55 error from ncredf (enter define mode)
! rc = -56 error from ncedf (exit define mode)
```

REVISION HISTORY:

```
1998.07.30 Lucchesi Initial interface design.
1998.07.30 Lucchesi Initial coding.
1998.09.24 Lucchesi Changed error handling.
1998.09.28 Lucchesi Added support for multiple precisions
```

2.10 GFIO_PutCharAtt – Write a user-defined character attribute (Source File: gfio.F)

This routine allows the user to define a character (string) attribute in an open GFIO file.

INTERFACE:

```
subroutine GFIO_PutCharAtt ( fid, name, count, buf, rc )
```

USES:

```
Implicit NONE
include "netcdf.inc"
include "gfio.h"
```

INPUT PARAMETERS:

```
integer fid ! File handle
character*(*) name ! Name of attribute
```

OUTPUT PARAMETERS:

```
integer rc ! Error return code:
    ! rc = 0 all is well
    ! NetCDF Errors
    ! ------
! rc = -36 error from ncaptc/ncapt (global attribute)
    ! rc = -55 error from ncredf (enter define mode)
! rc = -56 error from ncedf (exit define mode)
```

REVISION HISTORY:

```
1998.07.30 Lucchesi Initial interface design.
1998.07.30 Lucchesi Initial coding.
1998.09.24 Lucchesi Changed error handling.
```

2.11 GFIO_GetAttNames – Get global attribute names (Source File: gfio.F)

This routine allows the user to get the names of global attributes.

INTERFACE:

```
subroutine GFIO_GetAttNames ( fid, ngatts, aname, rc )  USES: \\
```

```
Implicit NONE
include "netcdf.inc"
include "gfio.h"
```

INPUT PARAMETERS:

integer fid ! File handle

INPUT/OUTPUT PARAMETERS:

integer ngatts ! Expected number of attributes (input)
! Actual number of attributes (output if rc=-2)

OUTPUT PARAMETERS:

REVISION HISTORY:

```
1998.08.05 Lucchesi Initial interface design.
1998.08.05 Lucchesi Initial coding.
1998.09.24 Lucchesi Changed error handling.
```

2.12 GFIO_AttInquire – Get information about an attribute (Source File: gfio.F)

This routine allows the user to get information about a global attribute of an open GFIO file. This is most useful for determining the number of values stored in an attribute.

INTERFACE:

```
subroutine GFIO_AttInquire ( fid, name, type, count, rc )

USES:

Implicit NONE
  include "netcdf.inc"
  include "gfio.h"
```

INPUT PARAMETERS:

integer fid ! File handle
character*(*) name ! Name of attribute

```
integer type
                 ! Code for attribute type
                 ! 0 = integer
                    1 = real
                     2 = character
                 ! 3 = 64-bit real
                 ! 4 = 64-bit integer
                 ! -1 = other
integer count
                 ! Number of items (length of array)
integer rc
                 ! Error return code:
                 ! rc = 0 all is well
                 ! NetCDF Errors
                 ! -----
                 ! rc = -58 error from ncaing
```

!NOTES: The returned integer "type" for 64-bit integer is not supported in the current implementation of netCDF/HDF. When a user writes a 64-bit integer attribute using PutIntAtt, it is actually saved as a 64-bit real by the HDF library. Thus, upon reading the attribute, there is no way for HDF/GFIO to distinguish it from a REAL number. The user must realize this variable is really an integer and call GetIntAtt to read it. Even for a 64-bit integer, type=4 will never be returned unless there are changed to HDF/netCDF.

REVISION HISTORY:

```
1998.07.30LucchesiInitial interface design.1998.07.30LucchesiInitial coding.1998.09.24LucchesiChanged error codes, added type assignment.
```

2.13 GFIO_GetIntAtt - Read a user-defined integer attribute (Source File: gfio.F)

This routine allows the user to read an integer attribute from an open GFIO file.

INTERFACE:

```
subroutine GFIO_GetIntAtt ( fid, name, count, buf, rc ) USES: \\ Implicit NONE include "netcdf.inc" include "gfio.h"
```

INPUT PARAMETERS:

```
integer fid ! File handle
character*(*) name ! Name of attribute
```

INPUT/OUTPUT PARAMETERS:

```
integer count ! On input: Number of items in attribute ! On output: If rc = -1, count will contain ! the correct count of this attribute
```

REVISION HISTORY:

```
1998.07.30 Lucchesi Initial interface design.
1998.07.30 Lucchesi Initial coding.
1998.09.29 Lucchesi Changed error handling. Added 64-bit support.
```

2.14 GFIO_GetRealAtt – Read a user-defined real attribute (Source File: gfio.F)

This routine allows the user to read a real attribute from an open GFIO file.

INTERFACE:

```
subroutine GFIO_GetRealAtt ( fid, name, count, buf, rc )
```

USES:

```
Implicit NONE
include "netcdf.inc"
include "gfio.h"
```

INPUT PARAMETERS:

```
integer fid ! File handle
character*(*) name ! Name of attribute
```

INPUT/OUTPUT PARAMETERS:

```
1998.07.30 Lucchesi Initial interface design.
1998.07.30 Lucchesi Initial coding.
1998.09.29 Lucchesi Changed error handling. Added 64-bit support.
1999.08.23 Lucchesi Changed .OR. to .AND.
```

2.15 GFIO_GetCharAtt – Read a user-defined character attribute (Source File: gfio.F)

This routine allows the user to read a character attribute from an open GFIO file.

INTERFACE:

USES:

```
subroutine GFIO_GetCharAtt ( fid, name, count, buf, rc )
```

```
Implicit NONE
include "netcdf.inc"
include "gfio.h"
```

INPUT PARAMETERS:

```
integer fid ! File handle
character*(*) name ! Name of attribute
```

INPUT/OUTPUT PARAMETERS:

```
integer count ! On input: Number of items in attribute
! On output: If rc = -1, count will contain
! the correct number of attributes
```

```
! rc = -2 type mismatch
! NetCDF Errors
! ------
! rc = -36 error from ncaptc/ncapt (global attribute)
! rc = -51 error from ncagtc/ncagt (global attribute)
```

1998.07.30	Lucchesi	Initial	interface design.
1998.07.30	Lucchesi	Initial	coding.
1998.09.29	Lucchesi	Changed	error handling.